By managing tasks like graphics generation and CRT refreshing, a dedicated VLSI display controller simplifies the design of intelligent graphics work stations.

Dedicated VLSI chip lightens graphics display design load

The role of graphics is becoming increasingly important for unscrambling the communications traffic between people and computers. Thanks to microprocessors and dedicated control ICs, designing high-reliability graphics work stations is now easier and less expensive than in the days of small-scale integration and expensive discrete-circuit CRT technology. Microprocessors simplify workstation design by transferring some graphics control tasks from hardware to software. However, a dedicated VLSI controller such as the 82720—with an on-board graphics processor—can push another step forward toward fast and economical design of high-quality intelligent graphics systems.

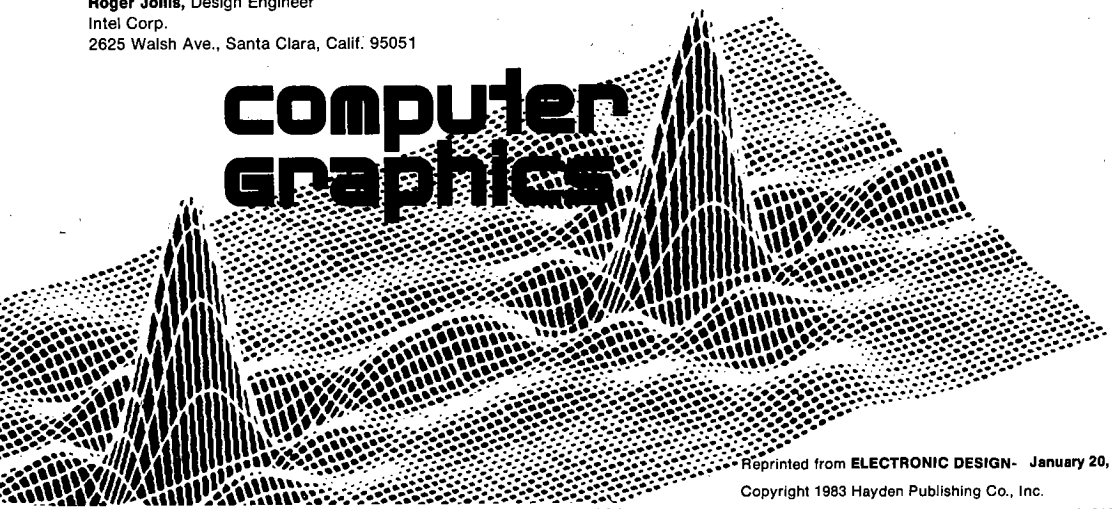
A typical application for the controller is a graphics work station aimed at high-end business and low-end engineering systems. Since such a station usually fits on the top of a desk, all of the electronics must be contained within a single printed-circuit board. This type of system requires a resolution of about 512 by 512 pixels and is frequently called on to display three-dimensional objects in various perspectives. To minimize the distortion of rotating objects, horizontal and vertical pixels should be equally spaced.

A typical display (500 vertices) must be drawn on the screen in less than 1 second to provide satisfactory interaction with the operator. The display may consist of lines, arcs, filled areas, and colors—seven colors are acceptable (see "A Look into Graphics Fundamentals").

Serial link interfaces station

An intelligent work station usually interfaces with a mainframe host via a serial communications link, a keyboard, and a serial link with an optional graphics tablet. This type of graphics input/output subsystem is diagrammed in Fig. 1. Two 5¼-in. floppy disks can satisfy the mass-storage needs of the system. Disk formatting must be compatible with the requirements of an IBM personal computer. Moreover, general-purpose software written for

Gary DePalma, Field Applications Engineer
Mark Olson, Product Marketing Engineer
Roger Jollis, Design Engineer
Intel Corp.
2625 Walsh Ave., Santa Clara, Calif. 95051



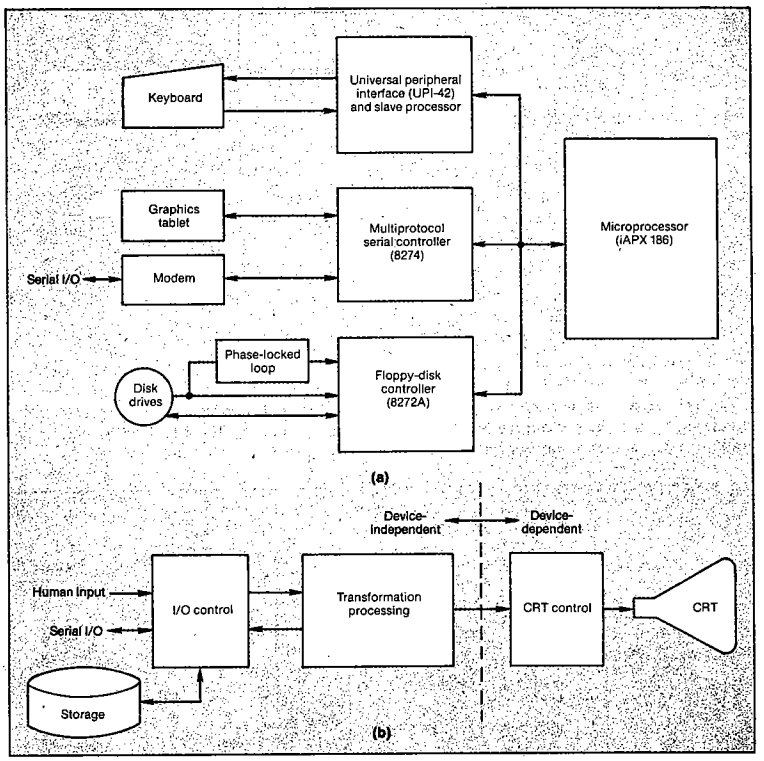
Reprinted from **ELECTRONIC DESIGN- January 20, 1983**

Order Number: 231310-001

this computer must also be able to run on the work station.

Two of the most basic functions of a graphics system are generating and refreshing images on the CRT screen. Information pertaining to the images is stored in the bit-map memory, where monochrome pixels are represented by single bits and color pixels by groups of bits. Lines and arcs defined in normalized screen coordinates must be converted into images of the physical object.

In a bit-mapped raster graphics system, lines described by a transformed display list are reduced to a series of dots and placed in the image memory. The selection of the dots that will be activated is achieved through a scanning conversion algorithm, which must create lines that appear very smooth, start and end as expected, and look symmetrical no matter in which direction they are drawn. The algorithm is repeated thousands of times to draw a single picture and thus must operate as quickly as possible. At the same time, the image in memory must be repainted on the screen 30 times/s for

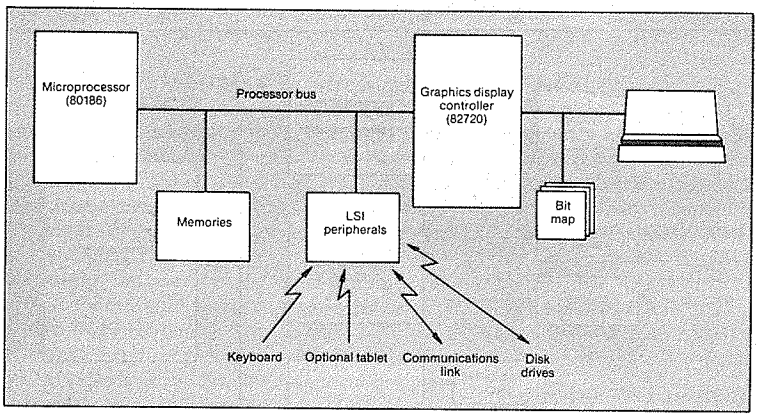


1. A graphics I/O subsystem for an intelligent work station consists of input peripherals (a keyboard and tablet), a serial communications link, and mass storage (floppy disks). Intelligence is provided by the microprocessor and the peripheral and memory controllers (a). The three basic tasks performed—I/O, transformation processing, and CRT control—all require data in the form of display lists stored in a data base (b).

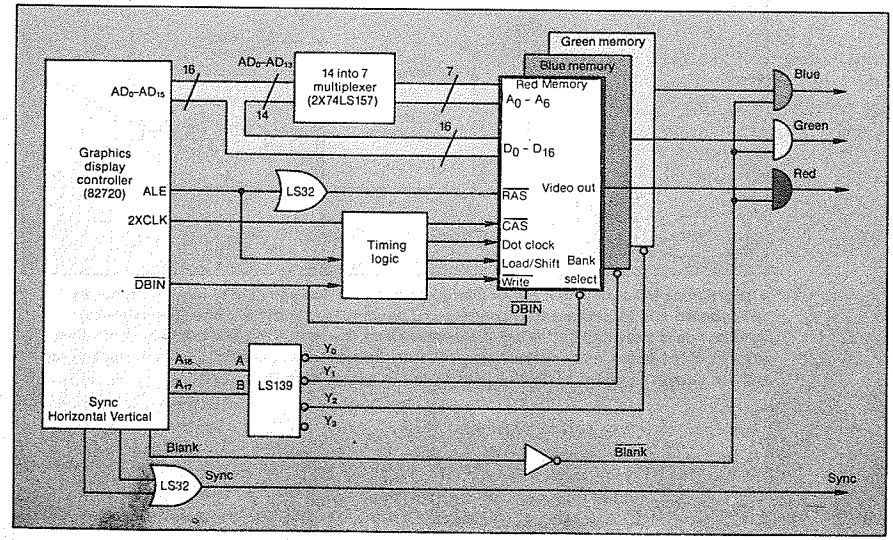
interlaced frames and 60 times/s for noninterlaced frames. Simple tasks, they nevertheless demand a high memory bandwidth.

Unlike other system control tasks, generating graphics figures requires both bit-manipulation and mathematics capabilities. Integer addition and multiplication operations calculate the coordinates of points on a line or a circle. But since pixels generally are neither complete words nor bytes, logical operations must be performed on the bits within the word that contains the selected pixel.

The inner loop of a so-called Bresenham line-drawing algorithm requires two or three addition operations, two comparisons or tests, and the masking of the correct value into the word for each pixel. Algorithms for drawing circles or filling areas are even more complex. In the inner loop of a filling algorithm, for example, the old word must be read from the bit map to determine whether some, all, or none of the pixels are within the area to be filled. If they are, the algorithm tests whether the pixels must be modified and then returns the word to the



2. The 82720 graphics display controller separates the tasks of graphics generation and CRT refreshing from other system tasks. That permits much greater system bandwidth, leading to graphics work stations that not only draw sharp pictures, but also offer color.



3. Three memory planes are implemented in the interface between the bit map and the graphics display controller. Three primary colors—red, green, and blue—are provided, with the controller's upper address bits responsible for selecting the memory planes during read/modify/write cycles.

231310-001

bit map. Because such algorithms are heavily exercised, they must execute at extremely high speeds to avoid an adverse impact on the system's overall efficiency.

Memory bandwidth is the most precious commodity in a graphics system. In this application, screen refreshing requires that 750,000 bits be read 60 times/s, equating to a bandwidth of almost 6 Mbytes/s. The picture refreshing, therefore, has the highest-priority access to memory because any missed readings show up as noise in the picture, a situation that sometimes occurs with simple systems possessing a single-microprocessor, single-memory scheme.

In the latter type of design, one processor handles all functions except refreshing, which is implemented by a discrete counter arrangement or a simple CRT controller chip. Nevertheless, the refresh memory bandwidth always slows down the microprocessor. That loss of speed can be eliminated simply by separating the processor's memory system from the bit map, a process that effectively doubles system memory bandwidth.

The 82720 graphics display controller can provide the means of separating graphics generation and CRT refreshing from the other tasks and also perform the two tasks quickly and concurrently with the others. Residing between the microprocessor and the bit-map memory and video logic, the controller refreshes the CRT like other CRT controllers, converts high-level commands into images by placing the proper data into the correct bit

A look into graphics fundamentals

The graphics data found in graphics display lists typically describes objects in the real-world Cartesian coordinate system conforming to the axes X, Y, and Z (see the figure). Graphics data does not take the form of one bit for every point on a line; rather it represents higher-level forms such as the end points of a line and

```
WALL: OBJECT
      MOVE TO [X, Y, Z]
      DRAW LINE TO [X2, Y2, Z2]
      DRAW LINE TO [X3, Y3, Z3]
      DRAW LINE TO [X4, Y4, Z4]
      DRAW LINE TO [X1, Y1, Z1]
END WALL
ROOF: OBJECT
      •
      •
      •
HOUSE: OBJECT
      WALL AT [T1]
      WALL AT [T2]
      WALL AT [T3]
      WALL AT [T4]
      ROOF AT [T5]
      WINDOW AT [T6]
      WINDOW AT [T7]
      DOOR AT [T8]
END HOUSE
```

the starting, ending, and center points of an arc.

The coordinate system handles physical measurement units such as inches, feet, or meters, which are typically represented in a computer by 16- or 32-bit integers or by floating-point formats. Ultimately, complex graphics structures are stored in a data base in a hierarchical form consisting of lists of X, Y, and Z coordinates.

The first step in designing a CRT subsystem involves selecting the resolution and scanning rates. All conventional raster-scanning monitors have a display area that is wider than it is high in the ratio of 4 to 3 (called the aspect ratio). For pixels to be square—equally spaced in both the X and the Y direction—the number of horizontal pixels must be 4/3 the number of vertical pixels. This is expressed as 4H-3V, where H and V represent the number of horizontal and vertical pixels respectively. Resolution depends on the total quantity of pixels, which must be a power of two. If it is not, the number of pixels must be rounded to the next highest power of two, in which case some bits will be wasted. Furthermore, the number of horizontal pixels must be organized as an even number of 16-bit words.

To prevent wasted bits, the number of vertical and horizontal pixels are chosen as large as possible without exceeding a power of two. For the display in question, 512H by 512V = 2^{18} = 262,144 pixels. A screen format of 576H by 432V normally meets all requirements. The total number of pixels is then 248,832, and the ratio of horizontal to vertical pixels (576/432) is correct. Furthermore, the number of horizontal pixels makes exactly 36 16-bit words.

After figuring the aspect ratio, the format of the bit-map memory is the next item to be considered. The screen contains about 250,000 pixels, each of which can be either black or one of seven colors. These eight shades can be represented by three bits/pixel (2^3 = 8), meaning that the bit-map memory must handle about 750,000 bits. The organization of the memory, however, must be determined according to the various tradeoffs.

The entire memory must be accessed 60 times/s since that is the rate at which the image must be painted to prevent flickering. That equates to a refresh rate of 16.7 ms. As a rule of thumb, the monitor displays information 75% of the time and is blanked for retracing operations 25% of the time. Thus the whole memory must be read and sent to the CRT during a 12.5-ms interval (16.7 × 0.75), which constitutes the active

map, and interfaces easily and simply with proprietary microprocessors.

The 82720 accepts high-level commands (such as DRAW LINE, DRAW ARC, and FILL RECTANGLE) and executes them at much faster speeds than general-purpose microprocessors, primarily because it is a dedicated graphics hardware processor. Burst drawing rates as high as 1 pixel every 800 ns can be achieved. Screen refreshing is handled directly by the controller. The displayed portion of the bit-map memory can be configured to allow the display to be scrolled through memory in any direction. The horizontal and sync periods both are fully programmable, as is the position of the sync pulse in the blanking interval. Furthermore, the controller can be programmed to refresh low-cost dynamic RAMs. In the design being considered, the 82720 offloads the microprocessor from low-level graphics tasks, as shown in Fig. 2.

For the bit-map interface, the memory is implemented as three planes, each 16 kwords by 16 bits, with each plane driving red, green, or blue (Fig. 3). The upper address bits—A_{16} and A_{17}—select the memory planes during read/modify/write cycles but are ignored during screen refreshing cycles.

The graphics display controller generates the Row Address Strobe (RAS) signal for the dynamic RAMs, but the remaining timing signals must be supplied by external devices. These signals are produced by a state-machine timing generator consisting of a 4-bit counter and two flip-flops. The state machine synchronizes itself with RAS after

portion of a frame.

To meet these requirements, it is helpful to break the bit map into three planes of 432 by 576 bits. While the screen is being refreshed, data is read from the same address in each of the three planes and sent serially to the screen. The memories can then be arranged as three 16-kword-by-16-bit arrays, requiring a memory cycle time of 800 ns and consequently permitting the use of relatively slow, low-cost 16-kbit dynamic RAM chips.

When drawing graphics figures, memory can be treated as a single large plane divided into three primary colors: red, green, and blue. Thus the low-order memory could represent the color red; the middle-order memory, green; and the high-order memory, blue. Each primary color requires the setting of just 1 bit/pixel. However, a secondary color—cyan, yellow, or magenta—necessitates setting 2 bits/pixel. Therefore, drawing in a secondary color takes two memory cycles/pixel and is slower than drawing primary colors. If this creates system problems, additional hardware can be used to draw more than one plane at a time. However, in the system example, drawing speeds are not only met, but also exceeded without relying on extra hardware.

Starting with the vertical refresh rate of 16.67 ms/frame, the basic timing can be analyzed. From the 16.67-ms figure, subtract the 1.25 ms required by the monitor for its vertical blanking. That leaves 15.42 ms for scanning the 432 lines on the active portion of the display. Dividing 15.42 ms by 432 lines gives 33.5 μs/line, equivalent to a horizontal scan rate of 28 kHz.

Vertical retracing requires 7 μs/line, and the active portion of each line is 35.7 μs. Subtracting 7 μs from 35.7 μs leaves 28.7 μs. During this time 576 pixels are displayed, for a pixel period of 28.7 μs/576 or 49.8 ns. This corresponds to a dot clock rate of 20.07 MHz, which is chosen as the system's basic clock rate.

Display lists and commands pass from the I/O subsystem to a unit that executes the transformation tasks. Transformations are primarily mathematical operations performed on the display units. Depending on the command, this module edits display lists, organizes them for display on the screen, or edits the display-list data base. By editing a display list, objects in the physical coordinate system can be created, destroyed, moved, or changed. Transforming a display list into a form compatible with the display is necessary, as the data base can have an unlimited real-world coordinate system in three dimensions, but the CRT screen is limited to only two dimensions.

Transformation tasks place a heavy burden on a microprocessor. For instance, in a typical transformation, a matrix multiplication is performed for every point on an object's display list. In three dimensions, each point requires multiplying a 4-element vector by a 4-by-4 matrix. Some of the elements are always zero, but the operation still takes 13 multiplications and 9 additions. A two-dimensional display list requires 4 additions and 4 multiplications. A typical display can contain hundreds or thousands of lines, each of which has two end points. Therefore the speed of matrix multiplication significantly influences system performance.

The coordinate system supported by the design example is three-dimensional and employs 32-bit integers. The system CPU executes 32-bit integer matrix multiplications at high speed. In conjunction with the graphic display controller, the drawing task is offloaded from the CPU, which in turn maximizes the central-processor time that can be allocated for those matrix multiplications. Waiting time is now much lower than in conventional systems. However, if a system requires floating-point transformations, the best high-speed performance is achieved with the addition of a numerical coprocessor.

the 82720 has been initialized. Figure 4 shows the complete schematic for each plane of the bit-map interface.

The remainder of the hardware design interfaces the graphics display processor, the processor memory, and the other peripherals with the 80186 microprocessor. The task is simplified by the processor's on-board chip-selection logic and wait-state generators. Furthermore, because of the processor's highly integrated architecture, the size of the overall hardware is quite small.

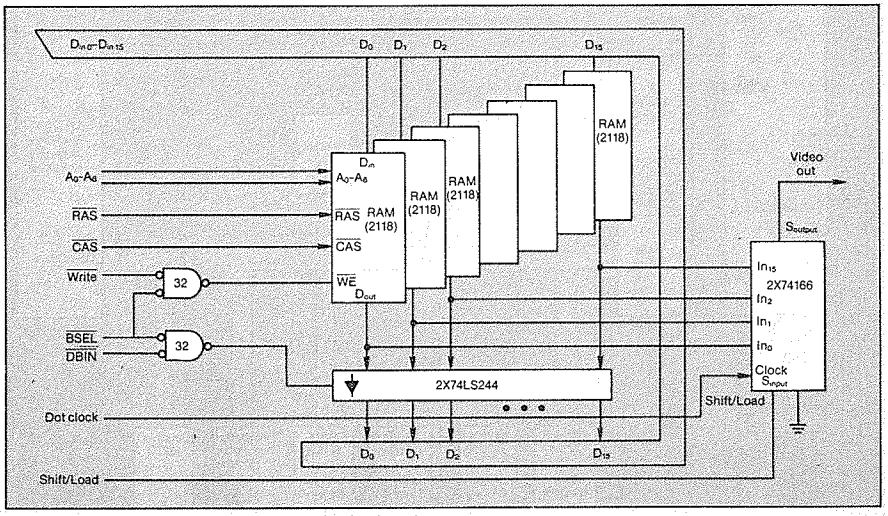
Joining processor and controller

Connecting the graphics display controller to the microprocessor is a simple task, as the processor's Data, Read, and Write signals are completely compatible with those of the 82720. However, because the controller has no chip-selection input, the Read or Write signals must be qualified through external hardware.

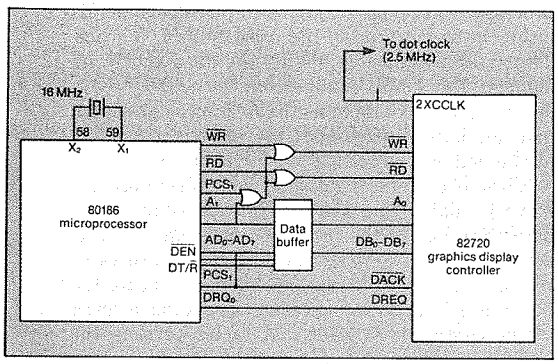
A number of chip-selection lines on the microprocessor can be programmed to place peripherals either in memory or in the processor's I/O space. Two gates are added to qualify the Read and Write signals. The DMA channel on the 80186 uses a second chip-select input as the Acknowledge signal, and data buffers are used to prevent bus contention at the end of a processor read cycle (Fig. 5).

Without buffers, the display controller must remove its data from the multiplexed address and data lines before the processor puts out the next address. At an 8-MHz clock rate, the processor requires that peripherals and memory vacate the bus in less than 85 ns; however, the standard speed of the controller is 100 ns. A faster version, the 82720-1, can be used, but it requires faster memory chips. A more cost-effective solution is simply adding buffers, if board space permits.

Serial communications to both the host and the optional tablet are handled by a multiprotocol serial controller (the 8274), which takes care of the host's synchronous and the tablet's asynchronous



4. The bit-map memory interface contains three address planes (one of which is shown here) to complete the graphics system. The $\overline{RAS}$ signal for the RAMs is generated by the graphics display controller.

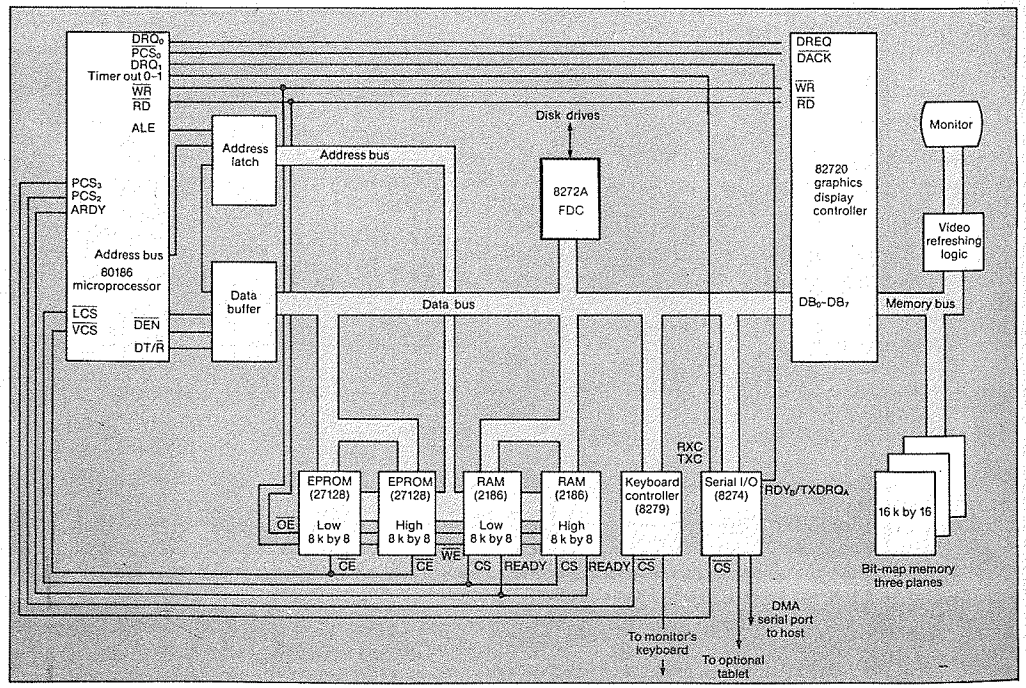


5. The interface between the 82720 and the system microprocessor is simple to implement because all of the processor's signals are compatible with the controller. It is necessary, however, to use external gates to qualify the RD or WR signals.

requirements. Interfacing is accomplished simply by connecting the buffered data bus, the latched lower-address lines, the Read and Write signals, and the chip-select. A final link brings the microprocessor's counter-timer output into the multiprotocol serial controller as a baud-rate clock. No buffering of the TTL support circuitry is necessary.

Universal chip interfaces keyboard

A universal peripheral interface chip (the UPI-42) serves as the keyboard interface and is programmed to scan the keyboard and interrupt the processor only on detection of a valid debounced keystroke. Mass-storage subsystems are managed by the 8272A floppy-disk controller. An external phase-locked loop circuit generates all of the timing signals reequired to connect a 5¼-in. drive to the system. On the microprocessor side, a DMA channel provides the link to the floppy-disk controller. Thus



6. A complete graphics control system is centered around an 80186 microprocessor and the 82720 controller. Local storage is provided by 32 kbytes of EPROM and 16 kbytes of RAM. The system comprises 85 chips and is housed on a single 12-by-12-in. printed-circuit board.

the processor has a high-speed disk interface, which loads it lightly.

To complete the graphics system illustrated in Fig. 6, 32 kbytes of EPROM and 16 kbytes of RAM support the microprocessor's program and display lists. The two EPROMs (27128s) come in 28-pin packages, thereby saving board space.

Hooking up the RAM chips is almost as straightforward. Since the microprocessor is a fully byte-addressable device, it can write bytes as well as words to the RAM. The chip-select input for the low (even) address RAM must be qualified with address A_0 at a logic zero, and the high (odd) address RAM must be qualified by the processor's Byte High Enable signal (BHE). The RAMs, designated 2186, have built-in controllers.

Since dynamic RAMs latch addresses on the leading edge of the chip-select signal, they must be qualified with the processor's Address Latch Enable signal to ensure that selection is made only after the address is valid. Then, a RAM latches the data to be written on the leading edge of the write pulse. The microprocessor's write signal must be delayed by one-half of a clock cycle to guarantee that data is valid at the correct time.

At this point, the design meets all of its performance goals. The system draws lines and circles at about 120,000 bits/s. That is approximately 82,000 pixels for a display consisting of even amounts of the three primary colors, as well as three secondary colors, and white. The 500 vectors of 25 pixels each can be drawn in about 0.15 s, six times faster than the 1-s requirement. The worst case—drawing all lines in white—can be accomplished in about one-third of a second. These specifications are satisfied when the graphics display controller is running from a 2.5-MHz clock. Drawing is performed only during retracing and the 82720 is programmed to use three memory cycles of each horizontal retrace for memory refreshing.

All of the components fit on a board measuring 12 by 12 in., so that the desktop size requirements are satisfied. The electronic components occupy about 100 in.^{2} of the low-cost, double-sided printed-circuit board. □

Bibliography:

Bresenham, J.E., "Algorithm for Computer Control of a Digital Plotter," IBM System Journal, 1965, 4(1) pp. 25-30.